

WHAT IS CLAIMED IS:

1 1. A method of managing space within an IMS database, wherein said
2 space is managed during loading or updating of said database, comprising:
3 after said IMS database is loaded, assuming control from an IMS program and
4 obtaining memory addresses of IMS control blocks built by said IMS
5 program and obtaining information about said IMS database;
6 passing control back to said IMS program;
7 when said IMS program attempts to insert data into said IMS database, either
8 during loading or updating of said database, reassuming control from
9 said IMS program;
10 selecting a storage location for said data to be stored;
11 modifying said IMS control blocks to indicate the storage location selected;
12 and
13 passing control back to said IMS program.

1 2. The method of Claim 1, wherein said storage location for said storage
2 data to be stored is selected based on IMS database user preferences.

1 3. The method of claim 2, wherein said data includes root segments and
2 said user preferences include storing said root segments in fixed storage locations.

1 4. The method of claim 2, wherein said user preferences include storing
2 said data in particular units of work.

1 5. The method of claim 3, wherein said data also includes non-root
2 segments and said user preferences include storing non-root segments in a block of
3 memory which also contains the root segment of said non-root segments.

1 6. The method of claim 5, wherein said non-root segments are stored in
2 said block of memory which also contains said root segment only if space is available.

[illegible]

1 8. A method of managing space within an IMS database, wherein said
2 space is managed during loading or updating of said database, comprising:
3 after said IMS database is loaded, assuming control from an IMS program and
4 obtaining memory addresses of IMS control blocks built by said IMS
5 program and obtaining information about said IMS database;
6 passing control back to said IMS program;
7 when said IMS program attempts to insert a first segment into said IMS
8 database, either during loading or updating of said database,
9 reassuming control from said IMS program, wherein said first segment
10 comprises a prefix component and a data component;
11 splitting said prefix component of said first segment from said data component
12 of said first segment, and appending a data link to each said prefix and
13 data component, said prefix component with said data link appended
14 becoming a second segment and said data component with said data
15 link appended becoming a third segment;
16 selecting storage locations for said second and third segments to be stored;
17 modifying said IMS control blocks to indicate the storage locations selected;
18 and
19 passing control back to said IMS program.

1 9. The method of claim 8, wherein said storage locations for said second
2 and third segments to be stored are selected based on IMS database user preferences.

1 10. The method of claim 9, wherein said second segment is either a root or
2 non-root segment, and wherein said user preferences include storing root segments in
3 fixed storage locations.

1 11. The method of claim 10, wherein said user preferences include storing
2 non-root segments in a block of memory which also contains the root segment of said
3 non-root segments, if space is available.

1 12. The method of claim 9, wherein said user preferences include storing
2 said second segment in a unit of work.

1 13. The method of claim 12, wherein if said second segment is a non-root
2 segment, it is stored in the unit of work which contains the root segment of said
3 second segment.

1 14. The method of claim 9, wherein said user preferences included storing
2 said second and third segments in different storage locations in one storage device.

1 15. The method of claim 9, wherein said user preferences include storing
2 said second and third segments in separate storage devices.

1 16. A method of managing space within an IMS database, wherein said
2 space is managed during loading or updating of said database, comprising:
3 after said IMS database is loaded, assuming control from an IMS program and
4 obtaining memory addresses of IMS control blocks built by said IMS
5 program and obtaining information about said IMS database;
6 passing control back to said IMS program;
7 when said IMS program attempts to insert data into said IMS database, either
8 during loading or updating of said database, reassuming control from
9 said IMS program;
10 selecting a storage location for said data to be stored;
11 storing said data in the storage location selected;
12 modifying said IMS control blocks to indicate the storage location selected;
13 and
14 passing control back to said IMS program.

1 17. A method of managing space within an IMS database, wherein said
2 space is managed during loading or updating of said database, comprising:
3 after said IMS database is loaded, assuming control from an IMS program and
4 obtaining memory addresses of IMS control blocks built by said IMS
5 program and obtaining information about said IMS database;
6 passing control back to said IMS program;
7 when said IMS program attempts to insert a first segment into said IMS
8 database, either during loading or updating of said database,
9 reassuming control from said IMS program, wherein said first segment
10 comprises a prefix component and a data component;
11 splitting said prefix component of said first segment from said data component
12 of said first segment, and appending a data link to each said prefix and
13 data component, said prefix component with said data link appended
14 becoming a second segment and said data component with said data
15 link appended becoming a third segment;
16 selecting storage locations for said second and third segments to be stored;
17 storing said second and third segments in the storage locations selected;
18 modifying said IMS control blocks to indicate the storage locations selected;
19 and
20 passing control back to said IMS program.

1 18. A program storage media readable by a machine and containing
2 instructions for performing the method contained in claim 1.

1 19. A program storage media readable by a machine and containing
2 instructions for performing the method contained in claim 8.

1 20. A program storage media readable by a machine and containing
2 instructions for performing the method contained in claim 16.

1 21. A program storage media readable by a machine and containing
2 instructions for performing the method contained in claim 17.